1993-09

Modeling and simulation of a fiber distributed data interface local area network

Schenone, Aldo Bruno.

Monterey, California. Naval Postgraduate School

(2)

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

MODELING AND SIMULATION OF A FIBER DISTRIBUTED DATA INTERFACE LOCAL AREA NETWORK

by

Aldo Bruno Schenone

September, 1993

Thesis Advisor: Dr. Shridhar Shukla

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)

2. REPORT DATE
September 23, 1993

3. REPORT TYPE AND DATES COVERED

4. TITLE AND SUBTITLE
MODELING AND SIMULATION OF A FIBER DISTRIBUTED DATA INTERFACE LOCAL AREA NETWORK

5. FUNDING NUMBERS

6. AUTHOR(S)
SCHENONE, Aldo Bruno

7. PERFORMING ORGANIZATION NAMES (S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
THE VIEWS EXPRESSED IN THIS THESIS ARE THOSE OF THE AUTHOR AND DO NOT REFLECT THE OFFICIAL POLICE OR POSITION OF THE DEPARTMENT OF DEFENSE OR THE U.S. GOVERNMENT.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

In this thesis, the performance of the fiber distributed data interface (FDDI) local area network .. evaluated by software simulation. The network is modeled with OPNET, a communication network simulation tool. The main focus of the model is on the medium access control (MAC) and the timing requirements that need to be met for the correct behavior of the protocol. Simulation data is presented to support results from previous analytic studies of distinctive features of the protocol, including the behavior of the token rotation time, synchronous frame delays, fairness of channel access for nodes transmitting asynchronous traffic, etc.
Comparisons between the simulated and theoretical results confirm the accuracy of the OPNET simulation tools and demonstrate that it may be used to model other protocols of particular interest.

14. SUBJECT TERMS
FDDI MAC, OPNET, protocols, LAN

16. PRICE CODE

15. NUMBER OF PAGES
55

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18, 298-102

Approved for public release; distribution is unlimited.

Modeling and Simulation of a Fiber Distributed Data Interface Local Area Network

by

Aldo Bruno Schenone
Lieutenant, Peruvian Navy

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September, 1993

Author: [redacted]

Aldo Bruno Schenone

Approved By: [redacted]

Shridhar Shukla, Thesis Advisor

[redacted]

Amr Zaky~~, Second Reader~~

[redacted]

Michael A. Morgan, Chairman
Department of Electrical and Computer Engineering

TABLE OF CONTENTS

I. INTRODUCTION

A. PROBLEM

Data communication and information systems are used extensively in maritime systems and are critical for a modern nation's warmaking capability. Information systems are vital in developing logistical and infrastructure bases and in coordination of operational activities.

As computer and computer-based equipment become more prevalent in maritime applications, the necessity to transmit and receive data to and from other shipboard system or shore-based networks becomes more apparent. Today, there is no general protocol profile that can be used to accomplish the communications requirements between maritime services and also to allow end-to-end interconnection through the terrestrial network.

The arrival of new applications (e.g., Differential Global Position System Transmissions, Vessel Traffic Systems, etc.) demand the capability to share information among a group of mobile and fixed entities. To this end, a global communication fabric connecting all mobile and static naval entities is necessary.

There are different standard communication protocols, and on a ship, a local area network may be implemented with any of them. Our interest is in performance evaluation of networks that are likely to be typical components of a global network by means of a simulation tool. The component network we have selected is a Fiber Distributed Data Interface (FDDI) local area network.

Figure 1 shows the intended application of an FDDI LAN deployed on a naval platform.

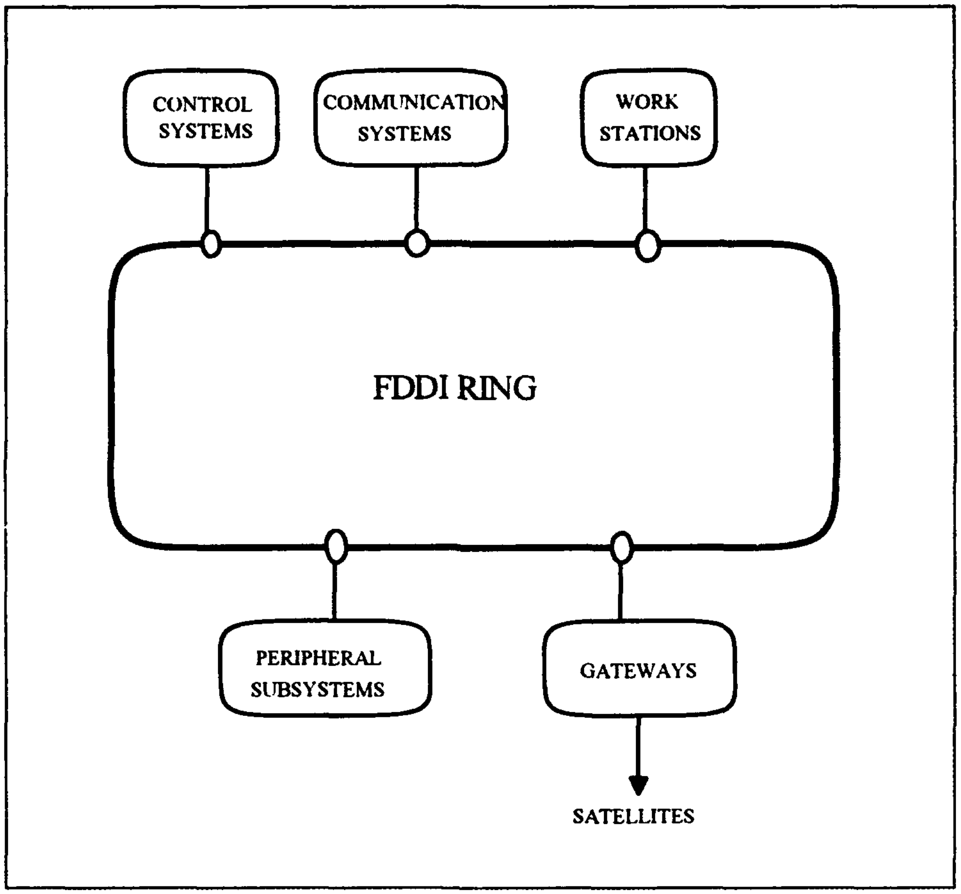

FIGURE 1. Example of FDDI Network.

B. SCOPE

The scope of this thesis is to:

- Conduct a performance analysis of the FDDI standard, a protocol that may be used to implement local area networks on each of the mobile entities, utilizing OPNET simulation software.
- Validate the applicability of the simulation software comparing the different measures of performance with theoretical results.

C. BENEFITS

The principal contribution made by this thesis is to model a simple FDDI configuration in OPNET, and thereby, gain understanding of the internal working of this tool. This thesis is concerned in determining the utility of OPNET in modeling a global grid.

The OPNET experience gained is expected to be helpful for the modeling of other communication protocols.

D. ORGANIZATION

The thesis is organized as follows. Chapter II reviews the FDDI protocol. The main focus is on the Media Access Control (MAC) and the timing requirements that need to be met for the correct behavior of the protocol. Chapter III reviews the OPNET organization and the capability of the simulation tool. An explanation of the FDDI protocol modeled, how the model was created, and for what particular simulation study it is applicable is also given. Chapter IV includes the simulation results obtained using this FDDI model. Chapter V provides the concluding remarks.

II. FDDI PROTOCOL DESCRIPTION

A. INTRODUCTION

The fiber distributed data interface (FDDI) is an American National Standards Institute (ANSI) standard for a 100 Mbps fiber-optic token ring [Ref. 5]. The proposed standard is designed to be effective over total path lengths of up to 200 km with 1000 physical connections. It is organized as in Figure 2.

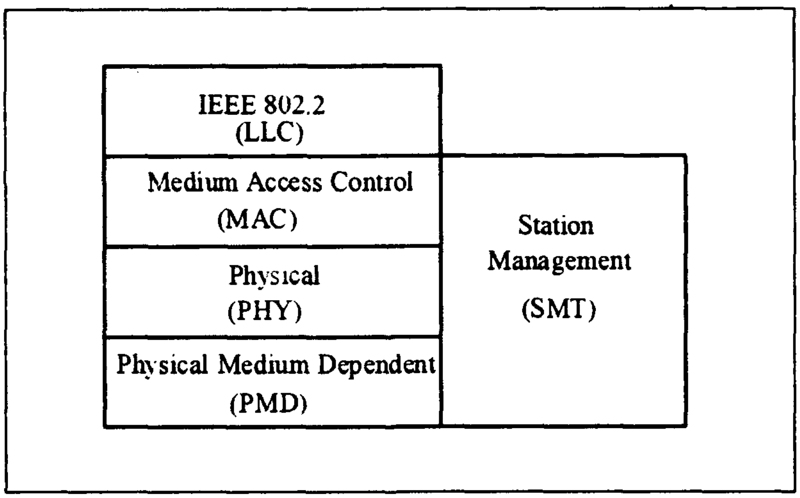

Figure 2. FDDI Protocol Organization [Ref. 5]

Like the IEEE 802.5 standard, FDDI employs the token ring algorithm. There are, however, two key differences that are intended to allow FDDI to take advantage of the high speed (100 Mbps) of its ring and maximize efficiency:

(1) As soon as a token frame is recognized, a waiting station absorbs it and after reception is finished, the station begins transmitting one or more data frames.

(2) A station that has been transmitting data frames releases a new token as soon as it complete the last data frame transmission, even if it has not begun to receive its own transmission back.

B. FDDI PROTOCOL

The FDDI standard encompasses both the MAC and physical layers (PHY), and supports the use of IEEE 802.2 logical link control (LLC).

Figure 2 describes the general FDDI protocol architecture, below the LLC level, which consists of four parts:

a. MAC: Portion of the data link layer that regulates access to the LAN medium.

b. Physical (PHY): Medium-independent portion of the physical layer, which includes the encoding of digital data.

c. Physical medium dependent (PMD): The medium-dependent aspects of the physical layer.

d. Station management (SMT): Provides the control to manage the process in the various FDDI layers at the station level.

1. Medium access control (MAC)

The main function of its Medium Access Control (MAC) is to schedule and perform data transfers to the ring. It follows a timed token rotation protocol [Ref. 2].

Each station has a set of timers that cooperatively attempt to maintain a specified token rotation time by using the observed network load to regulate the amount of time that a station may transmit. This provides a natural bound on the time between successive receipts of the token at any given station. All traffic in FDDI is classified as either synchronous or asynchronous. Normally, the synchronous traffic has critical delivery time constraints and comes from applications that require guaranteed bandwidth, such as real-time applications or from stations generating long streams that require high throughput. The asynchronous traffic comes from applications that generate short, bursty traffic with modest throughput requirements.

Asynchronous traffic is sent only if the synchronous traffic load of the ring is light enough to support it.

Figure 3 gives an example of ring operation. After station S1 has seized the token, it transmits frame F1, and immediately transmits a new token. F1 is addressed to station S3, which copies it as it circulates past. The frame eventually returns to S1, which absorbs it. Meanwhile, S2 seizes the token issued by S1 and transmits F2 followed by a token. This action could be repeated any number of times, so that at any one time, there may be multiple frames circulating on the ring. Each station is responsible for absorbing its own frames based on the source address field.

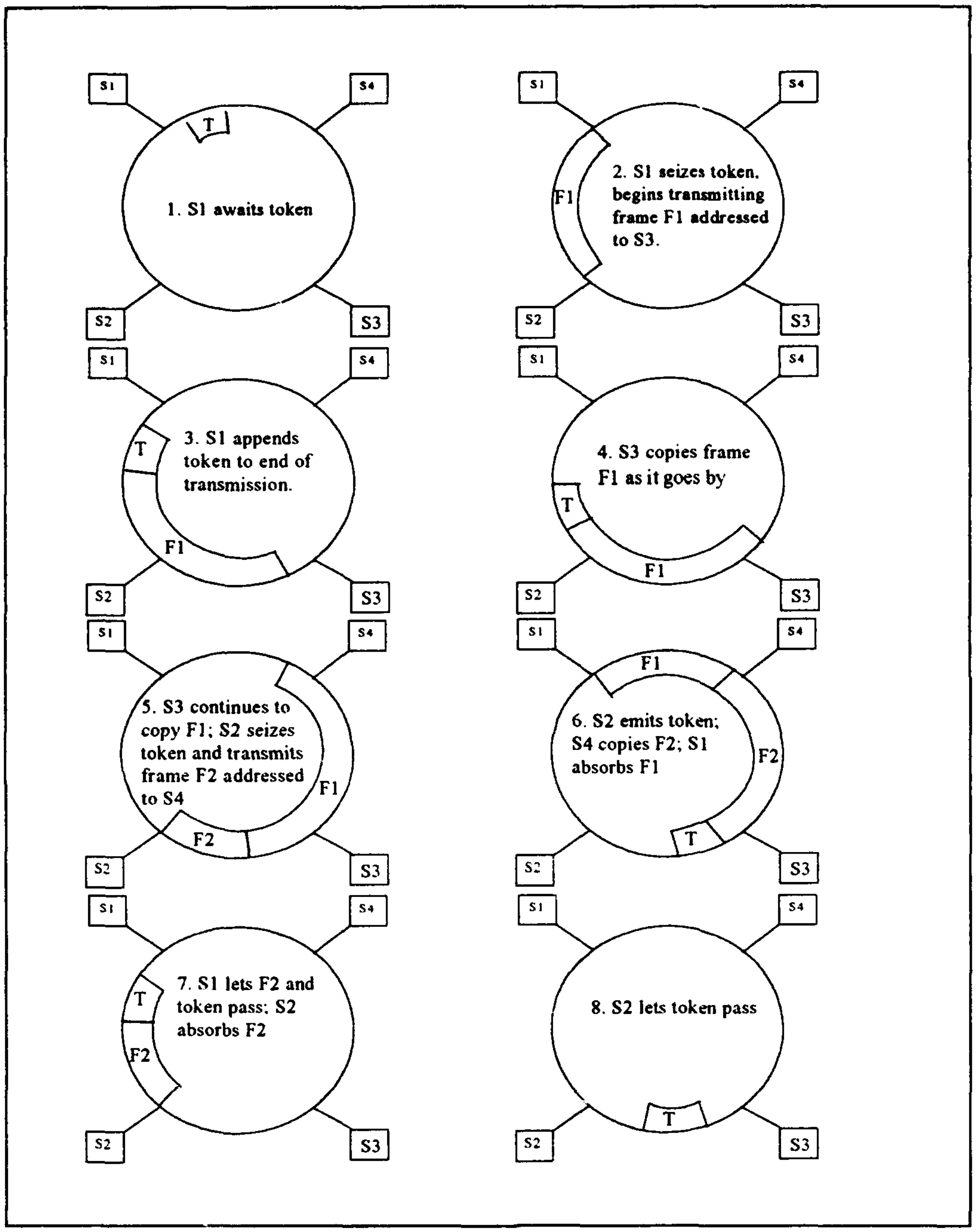

FIGURE 3. Example of FDDI token ring operation [Ref. 7]

Note that in FDDI, a station emits a new token immediately following the frame, whereas in IEEE token ring, a station emits the token only after the leading edge of its transmitted frame returns. The FDDI scheme is thus more efficient, especially in large rings.

To accommodate synchronous and asynchronous traffic, the FDDI scheme works as follows. A target token rotation time (TTRT) is defined and each station stores the same value for TTRT.

Some or all stations may be provided by SMT a synchronous allocation (SA_i), which may vary among stations. The allocations must be such that:

$$DMax + FMax + TokenTime + \Sigma SA_i \leq TTRT$$

where

SA_i	=	allocation for station i
DMax	=	propagation time for one complete circuit of the ring
FMAX	=	time required to transmit a maximum-length packet (4500 octets)
TokenTime	=	time required to transmit a token

The synchronous allocation for each station is a fraction of the expected token rotation time (T_Opr) for its synchronous transmission. The total of all synchronous assignments is not to exceed the value of T_Opr. Thus, all stations have the same value of TTRT and have separately assigned values of SA_i. In addition, several variables that are required for the operation of the capacity-allocation algorithm are maintained at each station. These variables are: Token-rotation timer (TRT), Token-holding timer (THT), Late counter (LC).

Each station's TRT is initialized to TTRT. When it is enabled, it counts down until it expires at TRT=0. It is then reset to TTRT and enabled again. LC is initialized at zero and is incremented when TRT expires. Thus LC records the number of times, if any, that TRT has expired since the token was last received at the station. The token is considered to arrive early if TRT has not expired since the station received the token, that is, if LC=0. When a station receives the token, its actions will depend on whether the token is early or late. If the token is early, the station saves the remaining TRT time in THT; resets TRT, and enables TRT.

The station can then transmit according to the following rules:

1. It may transmit synchronous frames for a time SA_i.

2. After transmitting synchronous frames, or if there were no synchronous frames to transmit, THT is enabled. The station may transmit asynchronous frames only if THT is greater than zero.

If the station receives the token and the token is late, LC is set to zero and TRT continues to run. The station can then transmit synchronous frames for a time SA_i. The station cannot transmit any asynchronous frames.

This scheme is designed to ensure that the time between successive sightings of a token is of the order TTRT or less. A given amount of this time is always available for synchronous traffic and any exceeding capacity is available for asynchronous traffic. Because of random fluctuations in traffic, the actual token circulation time may exceed TTRT [Ref. 3].

There are two modes of operation for the FDDI token ring, restricted token mode and nonrestricted token mode. Nonrestricted token mode is the normal mode of operation; support for restricted token mode is optional. The only difference between the two modes of operation is the method of allocating bandwidth for asynchronous transmission. In restricted token mode, a station may transmit asynchronous traffic only if the frame destination address is the station address.

To illustrate the use of station variables in FDDI, an example from [Ref. 6] is reproduced here. Figure 4 displays the values of TRT, THT, and LC for a particular station. In this example, the TTRT is 100 milliseconds (ms). The station's synchronous capacity allocation, SA_i, is 30 ms. The following events occur:

1. A token arrives early. The station has no frames to send. TRT is set to 100 ms and begins to count down. The station allows the token to go by.

2. The token returns 60 ms later. Since TRT=40 and LC=0, the token is early. The station sets THT←TRT and TRT←TTRT, so that THT=40 and TRT=100. TRT is immediately enabled. The station has synchronous data to transmit and begins to do so.

3. After 30 ms, the station has consumed its synchronous allocation. It has asynchronous data to transmit, so it enables THT and begins transmitting.

4. THT expires, and the station must cease transmission of asynchronous frames. The station issued the token.

5. TRT expires. The station increments LC to 1 and resets TRT to 100.

6. The token arrives. Since LC=1, the token is late, and no asynchronous data may be transmitted. At this point, the station also has no synchronous data to transmit. LC is reset to 0 and the token is allowed to go by.

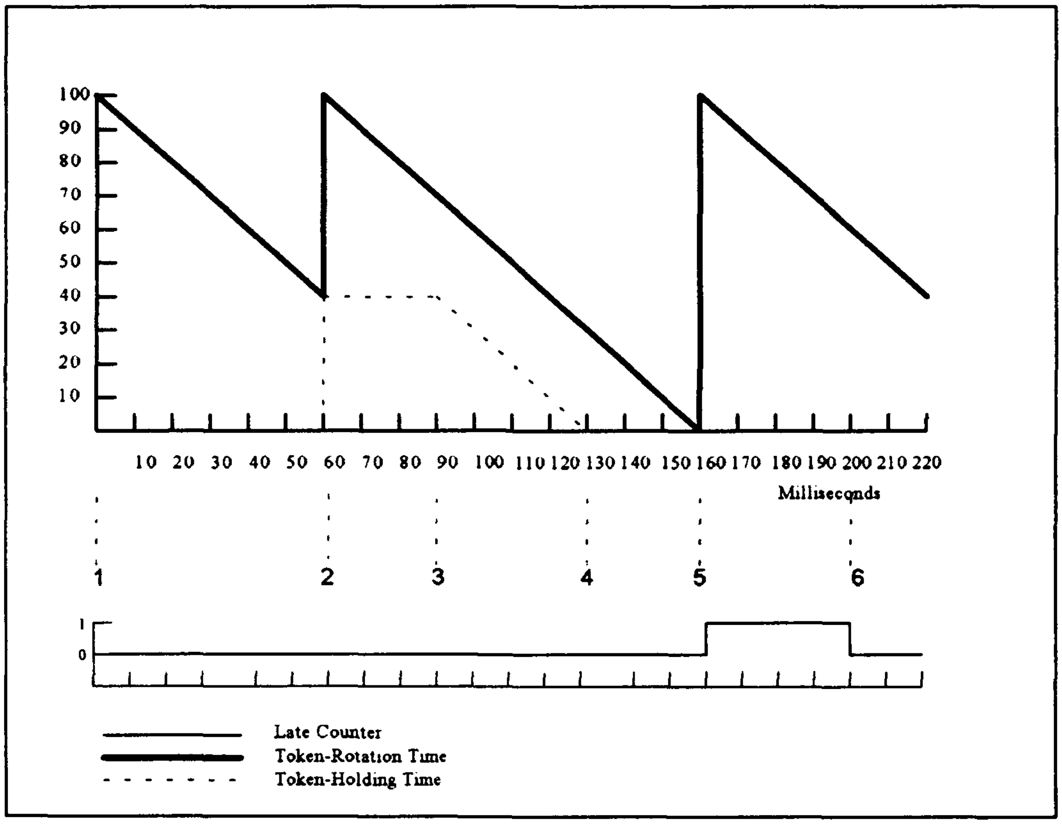

FIGURE 4 FDDI Capacity Allocation Example [Ref. 7]

Table 1 provides another example of a 4-station ring to illustrate the FDDI operation under the following assumptions: the traffic consists of fixed-length frames, TTRT=100 frame times, and SA_i=20 frame times for all stations. The total overhead during one complete token circulation is assumed 4 frame times, and all the stations are assume to always have a full backlog of synchronous and asynchronous traffic to transmit.

TABLE 1 OPERATION OF FDDI CAPACITY ALLOCATION SCHEME

Arrival/Departure time	Station Number	TRT	LC	Synchronous transmission	Asynchronous transmission
0	0	100	0	0	0
1	1	100	0	0	0
2	2	100	0	0	0
3	3	100	0	0	0
4	0	96	0	20	96
120	0	-16	1	—	—
121	1	-20	1	20	0
141	1	80	0	—	—
142	2	-40	1	20	0
162	2	80	0	—	—
163	3	-60	1	20	0
183	3	80	0	—	—
184	0	-80	1	20	0
204	0	80	0	—	—
205	1	16	0	20	16
241	1	64	0	—	—
242	2	0	0	20	0
262	2	80	0	—	—
263	3	0	0	20	0
283	3	80	0	—	—

TABLE 1 OPERATION OF FDDI CAPACITY ALLOCATION SCHEME (CONT.)

Arrival/Departure time	Station Number	TRT	LC	Synchronous transmission	Asynchronous transmission
284	0	0	0	20	0
304	0	80	0	—	—
305	1	0	0	20	0
325	1	80	0	—	—
326	2	16	0	20	16
362	2	64	0	—	—
363	3	0	0	20	0
383	3	80	0	—	—
384	0	0	0	20	0
404	0	80	0	—	—
405	1	0	0	20	0
425	1	80	0	—	—
426	2	0	0	20	0
446	2	80	0	—	—
447	3	16	0	20	16
483	3	64	0	—	—
484	0	0	0	20	0
504	0	80	0	—	—
505	1	0	0	20	0
525	1	80	0	—	—
526	2	0	0	20	0
546	2	80	0	—	—
547	3	0	0	20	0
567	3	80	0	—	—
568	0	16	0	20	16
604	0	64	0	—	—

The example begins after a period during which no data frames have been sent, so that the token has been circulating as rapidly as possible. In the table, the arrival and departure times of the token, the value of TRT at the time of arrival and departure, followed by the number of synchronous frames transmitted while the station holds the token, are shown for each station.

2. Physical Layer Specification

The physical layer specification for FDDI includes both medium dependent (PMD) and medium independent (PHY) parts. They are briefly summarized below.

(1) Physical Medium Specification (PMD): FDDI uses optical fiber with Light Emitting Diodes (LEDs) transmitting at a nominal wave length of 1300 nanometers. Connection between stations is made with a dual fiber cable employing a polarized duplex connector. The dimensions of the fiber cable are specified in terms of the diameter of the cladding layer that surrounds the core. The combination specified in the standard is 62.5/125 μm. The standard lists alternatives such as 50/125, 82/125 and 100/145 μm. In general, smaller diameters offer higher potential bandwidths but with higher connector loss.

The FDDI data link performance goals are 100 Mbps (125 Mbaud) through up to 2 Km of cable with a bit error rate of less than one in 2.5×10^{10} transmitted bits.

The error rate is determined by estimates of what is required to meet the overall network goals.

In [Ref. 1], W. E. Burr discuss the total structure of the FDDI standard, detailing information of the characteristics on the types of optical fiber waveguides commonly used.

(2) Data Encoding (PHY): Optical fiber is inherently an analog medium and signals can be transmitted only in the optical frequency range. The Amplitude Shift Keying (ASK) encoding technique is used to transform digital to analog signals; it is possible to use Frequency Shift Keying (FSK) and Phase Shift Keying (PSK) but they are difficult to implement at high data rate and the optoelectronic equipment would be too expensive and unreliable [Ref. 7]. The standard specifies the use of a code referred to as 4B/5B, encoded using Non Return to Zero Inverted (NRZI). In this code, a binary 1 is represented with a transition at the beginning of the bit interval and a binary 0 is represented with no transitions.

Stalling [Ref. 6] shows how the 4B/5B code is implemented, including the code group, the meaning of each symbol, and its assignment.

3. Stations and FDDI Network Configurations

Each FDDI station is composed of logical entities that conform to the FDDI standards. The role of a given station depends on the number of entities it has.

There are four different types of stations:

(1) Dual Attachment Station (DAS)

(2) Dual Attachment Concentrator (DAC)

(3) Single Attachment Station (SAS)

(4) Single Attachment Concentrator (SAC)

The configuration of each type is explained in [Ref. 7]. The definition of four station types allows for the creation of a wide variety of topologies like stand-alone concentrator with attached stations, dual ring, tree of concentrators, dual ring of trees, etc. Figure 5 shows a general FDDI topology that may be implemented with the four different station types.

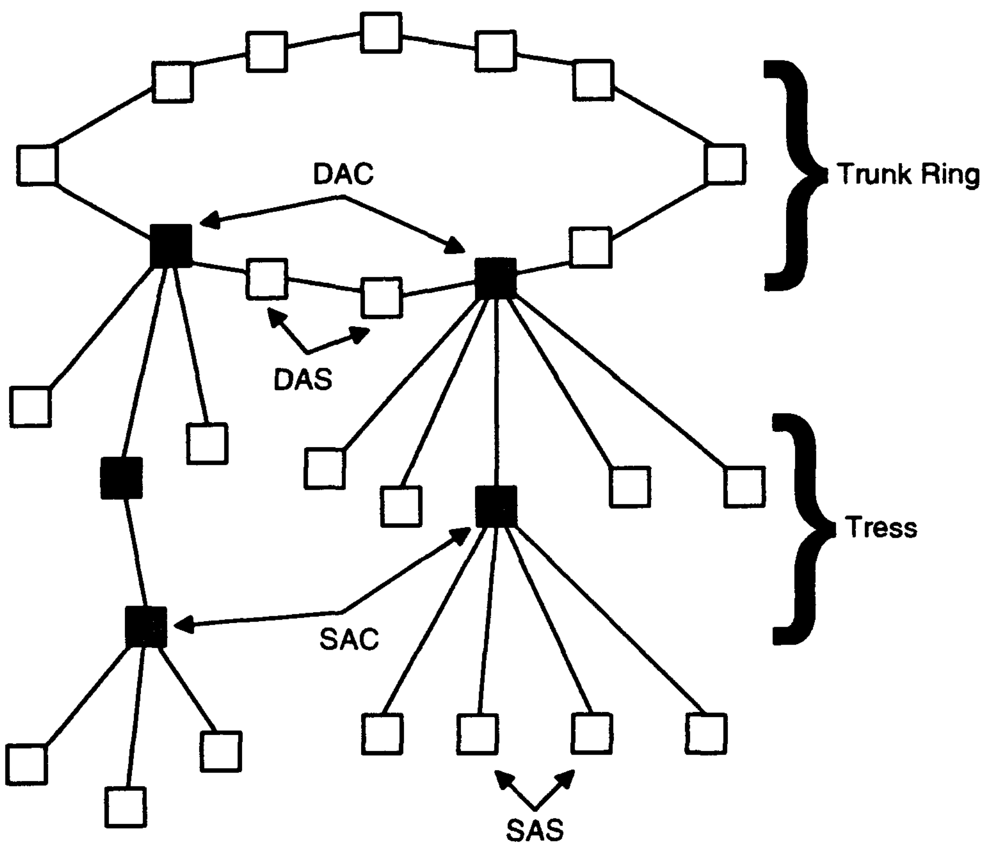

FIGURE 5. General FDDI Topology [Ref. 7].

4. Ring Monitoring

Each station is responsible for monitoring the functioning of the token ring algorithm. They monitors the ring for invalid conditions that require ring initialization. The base of these function is to keep track of the timers according to the FDDI MAC algorithm. Three processes are involved in error detection and correction:

- Claim token process
- Initialization process
- Beacon process

Additional details on ring monitoring may be found in [Ref. 5].

C. TIMING REQUIREMENTS

Station parameters which regulate the token rotation time must be initialized to satisfy the most stringent response-time requirements of all the stations, since each station's synchronous requirements will be different. The stations negotiate a value for the TTRT as part of the ring initialization process. Each station requests a value that is large enough to support its synchronous traffic needs. The shortest requested time is assigned to T_Opr, that parameter specifies the operational TTRT. T_Opr is an important ring parameter because it is used to limit the token rotation time.

1. Timed Token Protocol

Figure 6 shows the capacity allocation scheme. Each station's TRT is initialized to T_Opr; and it expires after a T_Opr period of time. Then TRT is reset to T_Opr and enabled again.

The FDDI MAC protocol provides for automatic recovery if a serious disruption of ring operation occurs. Ring recovery can be initiated by station management because of the expiration of the valid transmission timer, which protects against transient noise on the ring, or by expiration of the TRT when $LC \neq 0$. Ideally the last situation should occur only if the token is lost; it should not be triggered by slow token rotations.

Timed token rotation protocols dynamically adjust the amount of time that a station may hold the token to the speed of token rotation; if traffic on the ring is light, stations may transmit low priority frames; if traffic on the ring is heavy, stations may transmit only higher priority frames. As a result of this dynamic adjustment of access to the ring, the upper bound of the time between successive token arrivals (TRT) at a specific station is proved to be 2 x T_Opr [Ref. 3]. From that it was also proved that the token is guaranteed to return to a station within this period of time. The intuitive argument is that during each token rotation at most one T_Opr period of time can be used to transmit synchronous frames and at most one T_Opr period can be used to transmit asynchronous frames.

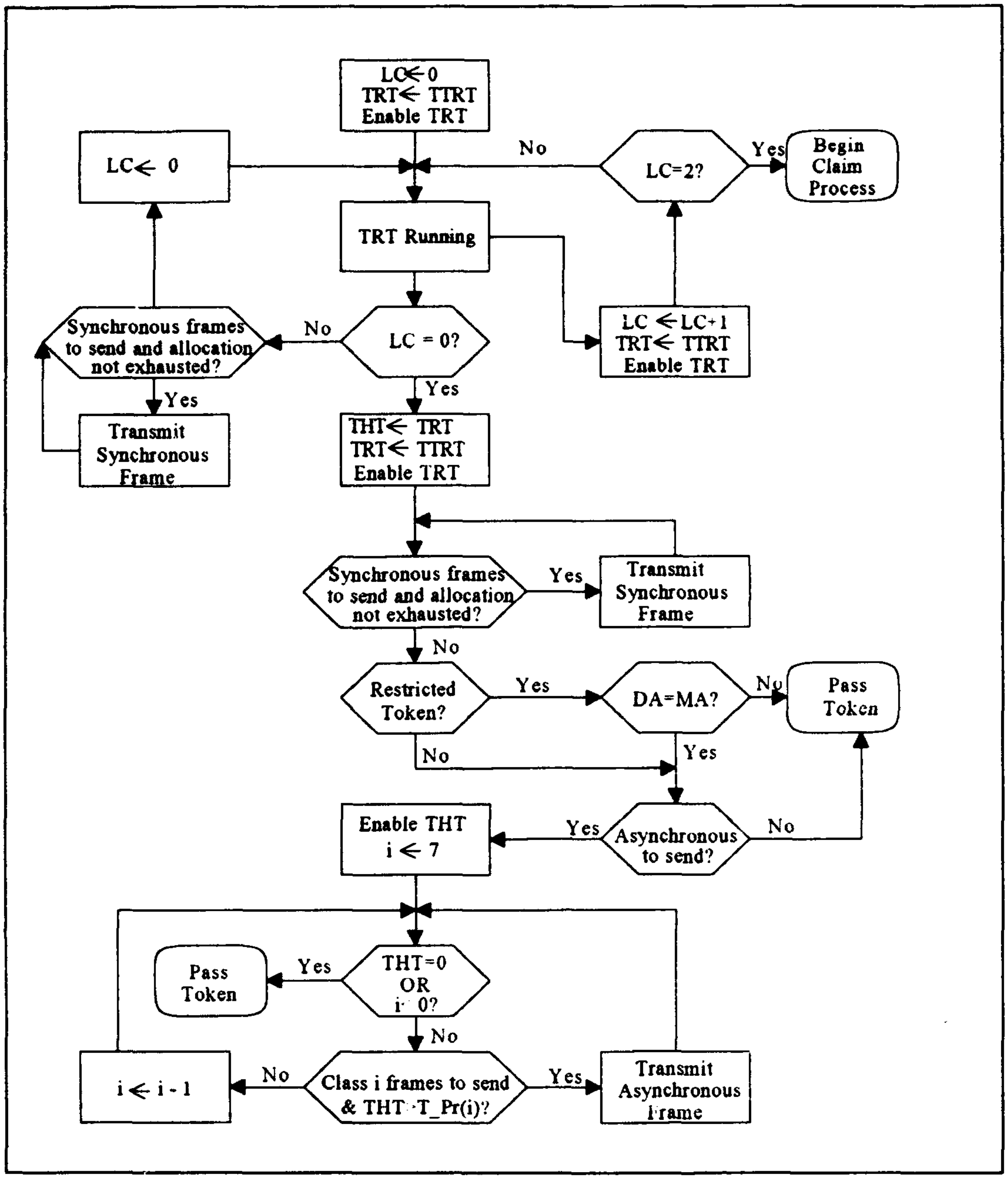

FIGURE 6. Capacity Allocation Scheme.

The T_Opr limit on asynchronous frames is obvious, since each station is allocated synchronous bandwidth as a percentage of T_Opr. The higher layer entity on the network responsible for making synchronous bandwidth assignments should ensure that the sum of all synchronous allocations does not exceed 100 percent of T_Opr.

However, the T_Opr limit on asynchronous transmission is not obvious. Since a station is allowed to complete transmission of an asynchronous frame, which was being transmitted when its THT expired, the limit on asynchronous transmission on a single station is T_Opr plus the amount of time required to transmit a frame of maximum length.

2. Single Asynchronous Priority Level

Dykeman [Ref. 2] has developed an equation relating the protocol parameters and the physical ring characteristics to the maximum total throughput for an FDDI token ring when only one asynchronous level is in use. The physical ring characteristics of his model include the ring latency and the number of active transmitting stations. Dykeman's work is based on the following assumptions: each active transmitting station continuously has frames queued for transmission, frame transmission times are of constant length and all transmissions beyond the expiration of the THT, owing to frame transmissions in progress, are of equal length. The derived generalized maximum-throughput expression with an arbitrary number of active stations using a single asynchronous priority level is:

$$Throughput_{\max} = \frac{(n \times tot\_tx\_time + n^2 \times tx\_window) \times tx\_rate}{n \times tot\_tx\_time + n^2 \times tx\_window + (n^2+2n+1) \times r\_lat} \qquad (1)$$

Where:

n = Number of actively transmitting stations.

r_lat = Total ring latency (signal propagation delay + sum of all station latencies)

R = Residual frame transmission time.

F = Frame transmission time.

$tx\_window = T\_Opr - r\_lat$

$tot\_tx\_time = \text{CEILING}(tx\_window/F) \times F$

If we take the limit of the above expression when the number of active transmitting stations goes to infinity, we find that:

$$\lim_{n \to \infty} throughput_{\max} = \frac{tx\_window \times tx\_rate}{tx\_window + r\_lat} = \frac{T{+}Opr - r\_lat}{T\_Opr} \times tx\_rate \qquad (2)$$

This limit is in agreement with the formula given by Ulm [Ref. 8] for a ring utilization of a timed token protocol.

III. FDDI OPNET MODEL

A. OPNET

1. OVERVIEW

Optimized Network Engineering Tools (OPNET) is a comprehensive engineering system capable of simulating large communications networks with detailed protocol modeling and performance analysis. The environment presents a modern graphical user interface which supports multi-windowing and makes use of menus and icons. OPNET is based on a series of hierarchically-related editors which directly can be used to simulate the structure of actual networks.

The lower layer for modeling is the Process Editor; it uses a state-diagram approach to support specification of any type of protocol, algorithm, or queuing policy. States and transitions graphically define the progression of a process in response to events. Within each state, general logic can be specified using the C language and a kernel procedure library. Users can construct entirely new process models, or modify those already provided.

Once the process model has been built and compiled, the next step is to implement the node model. The node editor graphically captures node architecture, which are diagrams or data flow between modules representing hardware and software subsystems. Module types include processors, queues, traffic generators, receivers, and transmitters. Processors are general modules that provide flexibility in protocol and algorithm specification.

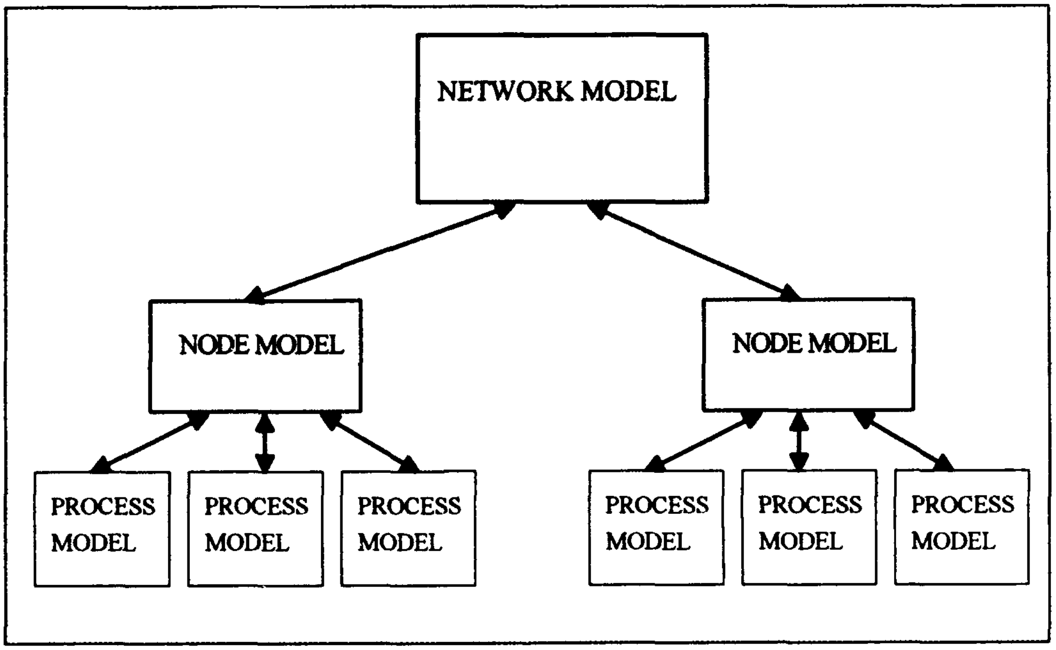

FIGURE 7. OPNET Organization Scheme.

Finally the network editor graphically captures the physical topology of a communications network. Networks consist of node and link objects, which are graphically assembled and characterized via menus. To create node objects, users select node types from a library of example and user-defined models. Each node has a specific set of attributes which are used to configure it.

Network models can be applied in a dimensioned workspace with a user defined scale. Users can load a customized cartographic background to setup a physical context for wide-area networks. Simulation execution normally involves the test of specific points of interest, that can be accomplished with the probe editor. Probes monitor statistics computed during the simulation. The simulation tool allows specification of a sequence of simulations which use particular input and output options.

Another capability of OPNET is to present statistics collected during simulations in the form of two dimensional graphs or text listings; by means of the analysis tool. The filter editor is used to define filters to mathematically process, reduce, and combine statistical data.

In summary, the eight opnet tools already described, fit into three categories as shown in Figure 8.

Model Development	Simulation Execution	Result Analysis
(1) Network	(5) Probe	(7) Analysis
(2) Node	(6) Simulation	(8) Filter
(3) Process		
(4) Parameter		

FIGURE 8. OPNET Tools Categories.

B. MODEL

1. MODEL RESTRICTIONS

The previous chapter described the basic mechanism and operational context of the FDDI MAC entity. This chapter discusses the implementation choices made in constructing an OPNET model of MAC. Because the model is intended for performance estimation, certain parts of the protocol have been simplified or omitted. It is important to understand which mechanism are modeled. The restrictions that the model presents are as follows:

a. The first restriction is that the ring initialization and recovery processes are not modeled specifically because its primary usefulness is in obtaining measurements of steady state performance. The initial alignment of station timers and command process by which all MAC entities negotiate the TTRT is performed in an essentially static manner. This also applies to the distribution of synchronous transmission bandwidth.

b. The second restriction has to do with the modeling of error conditions and in general, the role of the station management. The interface between MAC and SMT is not presently incorporated into the implementation of the MAC.

c. The last restriction is the simulation of different priorities while transmitting asynchronous traffic. Unless the model could be extended to provide these simulation, it is necessary to create some modifications to the code.

2. MODEL FEATURES

The FDDI model used in this thesis is based on the model applied by OPNET. It incorporates a simulation acceleration feature for modeling the passing token from station to station. Normally when the ring experiences an idle period with no transmissions, the token may be passed many times in short periods of time; consuming large amounts of real time while producing data which is of little interest. The incorporated acceleration procedure is employed to jump over these periods.

The model does incorporate the interfaces between MAC and LLC, as well as those between PHY and MAC. Besides the primary data transfer features of FDDI are explicitly modeled, including synchronous and asynchronous transmission. The effects of propagation delay and station latency are also incorporated into the model. The parameters which may be controlled by the user include the number of stations attached to the ring, asynchronous bandwidth allocation at each station, requested value of the TTRT by each station, address of the station which launches the token when the simulation begins, the delay incurred by frames and tokens as they traverse a station's ring interface, the propagation delay separating stations on the ring, the rate of exponentially distributed frame generation at each station and the size of generated frames, the mix of asynchronous and synchronous frames generated by each station, and the range of destination addresses for the frames generated at each station.

3. PROCESS LEVEL MODEL

There are three processes forming components of the FDDI model; the fddi_mac, fddi_gen and fddi_sink.

a. *fddi_mac*

Process which represents an FDDI MAC entity. This process model is specified as a finite state machine which manages the timers and state variables associated with a single MAC entity. The responsibilities of the fddi_mac process include forming MAC frames which encapsulate data received from LLC, repeating frames destined for other stations, decapsulating data from frames destined for this station and passing data to LLC, stripping frames originated by this station, maintaining THT and TRT timers, determining token usability, and transmitting MAC frames into the ring according to the rules discussed in the previous chapter of this thesis.

The state diagram of the fddi_mac process is shown in Figure 9; the function of each state is discussed below.

1) Claim: It is the initial state of the process. It is entered upon receipt of a begin simulation interrupt which is delivered by the simulation kernel as the simulation starts. In this state the initial negotiation with regard to the value of TTRT is emulated. All fddi_mac processes in the ring compare among them their requested value of TTRT (T_Req), by means of a global variable Fddi_T_Opr, which holds the lowest value requested.

2) Init: This state is entered by every fddi_mac process after the completion of the claim phase. Init is not reentrant because the actions achieved are related to initialization and do not need to be performed more than once. The assignment of values to state variables like the station latency, the inter-station propagation delay, and the combined inter-station delay are performed here. The process also define procedures for manipulating and querying timers, and acquires knowledge of its own station's address and places this value in the state variable my_address.

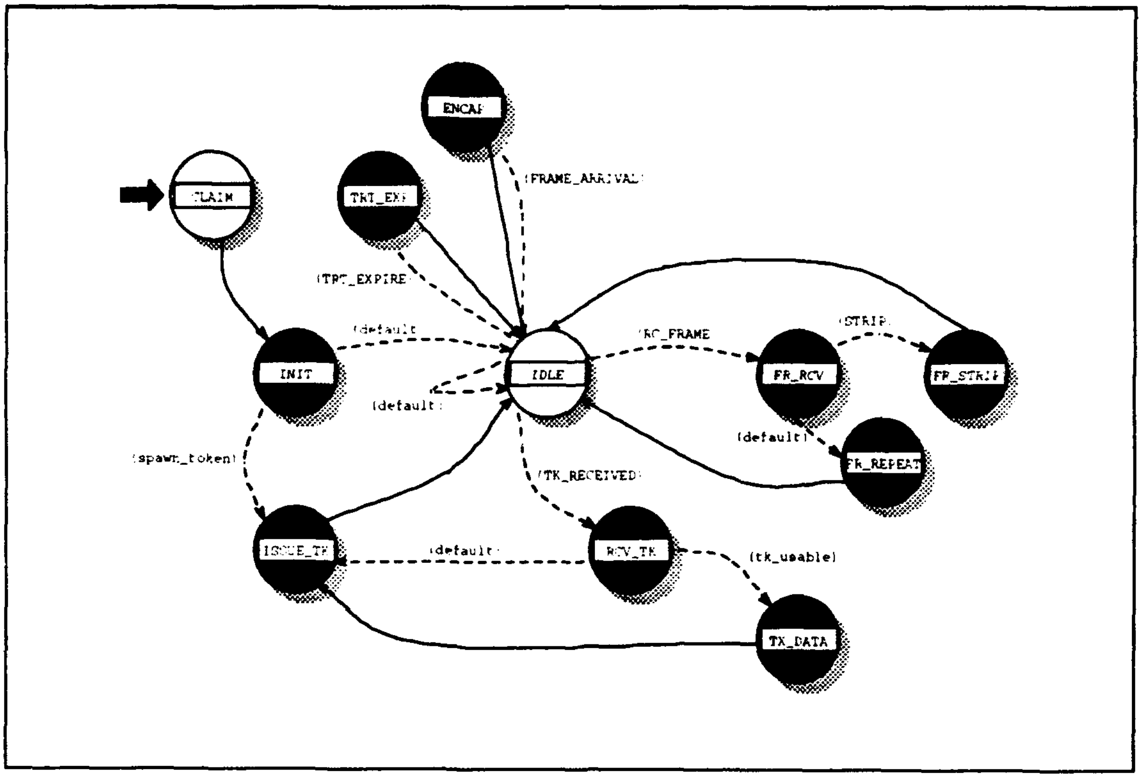

FIGURE 9. Fddi_mac Process State Diagram [Ref. 9].

3) Issue_Tk: This state is entered whenever fddi_mac needs to issue a token. Its primary action is therefore to forward the OPNET packet which represents the token.

4) Idle: This state is the main branching point for event processing in the steady state operation of fddi_mac. The interrupts to which the process will respond while in the Idle state are stream interrupts signaling the arrival of frames or tokens from PHY, self interrupts representing the expiration of TRT, and remote interrupts requesting that a token be generated and inserted into the ring. All interrupts are processed by leaving the Idle and going to the appropriate destination state.

5) TRT_Exp: This state is entered when fddi_mac receives a self interrupt indicating that the TRT timer has expired. At this point, the timer is reset to expire one TTRT into the future and a corresponding self interrupt is requested. Late_C is also incremented to indicate the lateness of the token.

6) Encap: The primary actions of this state are to acquire the arriving data and the associated control information when fddi_mac receives a service data unit from LLC; and use these to create a MAC frame suitable for transmission to a peer MAC entity via PHY. The resulting MAC frame is enqueued until a later time where it becomes eligible for transmission in the Tx_Data state.

7) Rcv_Tk: This state is entered upon receipt of a token from PHY. It obtain the class of the token (restricted or non-restricted), and according to each case, Rcv_Tk performs different actions adjusting parameters and variables according to the timing requirement specifications discussed in the previous chapter.

8) Tx_Data: This state is entered when the token is captured and a determination that it is usable is made. Its role is to dequeue and send frames into the ring until the token is no longer usable by this station, at which time the token is forwarded down-stream.

9) Fr_Rcv: In this state a frame has been received from PHY. The only action performed here is to extract the source address of the packet so that a determination can be made with regard to stripping the frame from the ring.

10) Fr_Repeat: In this state a frame originated by a station other than itself has been received. The frame will be repeated until it reaches its destination, so in the case where the frame's destination address does not correspond to the station which has received the message, the frame is repeated onto the ring, and the propagation delay and station latency are accounted for.

11) Fr_Strip: A frame originated by this station is received. The frame is discarded and fddi_mac returns to the Idle state.

b. fddi_gen process

It generates the frames to transmit in each station. The process has only two states; Init, where the process determines the identity of the own processor to use in finding attributes and setting up the exponential distribution with given mean for frames arrival generation; and the arrival state, where fddi_gen determines the characteristics of the generated packet, length, destination address, overall size, token class, and type of frame (synchronous or asynchronous).

c. fddi_sink process

It receives the frames addressed to the station, updates the variables used to measure the performance; and at the end of the simulation, it writes the scalar performance statistics and input parameters.

4. STATION NODE MODEL

The fddi_station model is the basic component of the FDDI network model. The model includes a traffic source, a traffic sink, a MAC entity, and a PHY entity. This entities are modeled in terms of modules which are provided by OPNET's node editor. The modules are connected via OPNET streams over which tokens and frames can be forwarded. The layout of the model is graphically depicted in Figure 10.

The phy_tx and phy_rx modules serve as the physical interface to the ring transmission medium. Frames and tokens are received by mac from phy_rx and forwarded from mac to phy_tx via stream 2 from/to the next up/down-stream neighbor.

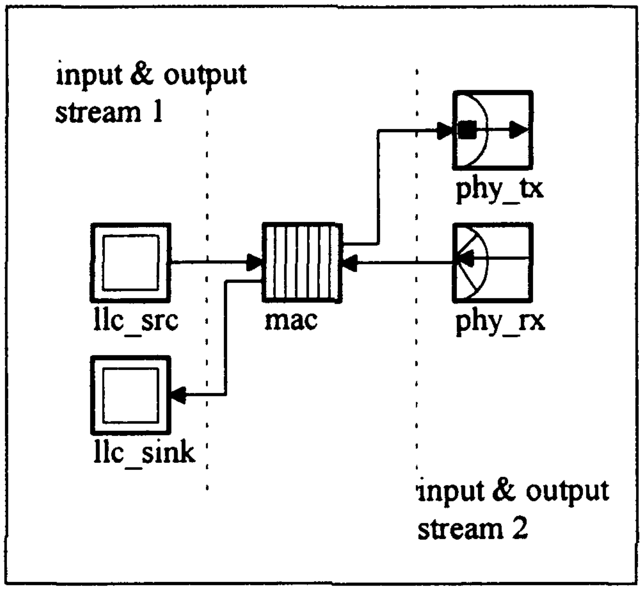

FIGURE 10. Fddi_station Node Model [Ref. 9].

The queue module mac occupies the place of the MAC entity in the station and has the responsibility of token and timer management, frame capturing and repetition, and queuing of transmission requests.

The processor llc_src provides frames for transmission; its behavior is prescribed by the fddi_gen process model described before. The processor llc_sink provides a receptacle for frames captured and addressed to the station, and forwarded by the mac processor.

5. NETWORK LEVEL MODEL

The ring topology and number of stations are the primary specifications comprised in the network level model, since the inter-station propagation delay has been made a global simulation attribute.

An important specification which occurs at the network level is the assignment of procedures which model the operation of the links. In our case, point-to-point links are used and their internal operation and transmission mechanisms are modified to reflect the actual behavior of FDDI interfaces.

Once the network is constructed; the Network Editor allows a number of attributes to be set for each station via menus. It is also possible to set those attributes via an environment file which is interpreted at the time of simulation.

6. SUMMARY OF MODEL CHANGES

The changes in the model were initially made to compensate for the original restriction to simulate priority transmission by asynchronous traffic.

a. Fddi_gen Process Code

The fddi_gen code is used to setup a distribution for generation of priority transmissions. In the case of this thesis, we create a uniform distribution from values between 0 and 7 in the INIT state. The saved value is then pointed to a location in memory by the pri_dis_ptr pointer variable.

At the arrival state, if the frame to transmit is asynchronous then the priority value previously generated is assigned to the dest_pri variable.

b. Fddi_mac Process Code

The T_Pri array which is initialized in the INIT state is used as follows. Before a frame is transmitted, the time remaining in THT timer is checked to see if its sufficiently large to permit the frame to be placed on the ring. The value used for this check is the value in T_Pri array corresponding to the frame's priority level; for instance, if a frame's priority is one, the model uses the value T-Pri [1]. If the remaining time is not large enough, the frame is returned to the input queue.

IV. SIMULATION RESULTS

In this chapter, we discuss the different outputs obtained with the FDDI network model and provide a comparison with the theoretical results.

A. STANDARD MEASURES OF PERFORMANCE

The ring configuration used to obtain the results in this section is presented in Table 2. The network is homogeneous, all traffic is asynchronous, and each node generates frames at the same specified arrival rate. Interarrival times for frames at each node are exponentially distributed.

1. MEAN FRAME DELAY

Figure 10 is a graph of the mean frame delay versus the offered load. The measured mean delay is the time from generation of the frame at the source node to reception of the frame at the destination node. This includes queuing delay at the source node, transmission time, and the time required for the frame to propagate from the source to the destination node. Delays due to processing at the upper layers of the OSI network architecture are not included.

TABLE 2. RING CONFIGURATION

Parameter	Value
Number of Nodes	20
Distance Between Nodes	30 meters
T_Opr	40 milliseconds
Frame Size	4000 bytes
Header Size	40 bytes
Station Latency	600 nanoseconds
Traffic Type	Asynchronous
Arrival Rate	154 frames/second

As a lower bound delay, the transmission time for a 4040-byte frame (frame + header) is :

$$4040\frac{bytes}{frame} \times 8\frac{bits}{byte} \times 10^{-8}\frac{\sec}{bit} = 323 \times 10^{-6}\sec$$

and the propagation delay is a maximum of 3.05 microseconds:

$$5.085\frac{micro\sec}{Km} \times 0.03Km \times 20Stations = 3.05micro\sec$$

The station latency corresponds to a 60 bit internal delay for each node, necessary to recognize what kind of frame the station is receiving:

$$Station\_Latency = 60bit \div 10^{8}bits/\sec\ = 0.6micro\sec$$

In Figure 11, the mean frame delay increases slowly as a function of the offered load. The slope of the curve does not increase rapidly until the offered load exceeds 95% of capacity, the average delay is approximately eleven frame-transmission times. Maximum frame delays experienced in these runs range from 600 microseconds at 5% offered load to 94 microseconds (approximately double the T_Opr value) at 98% offered load.

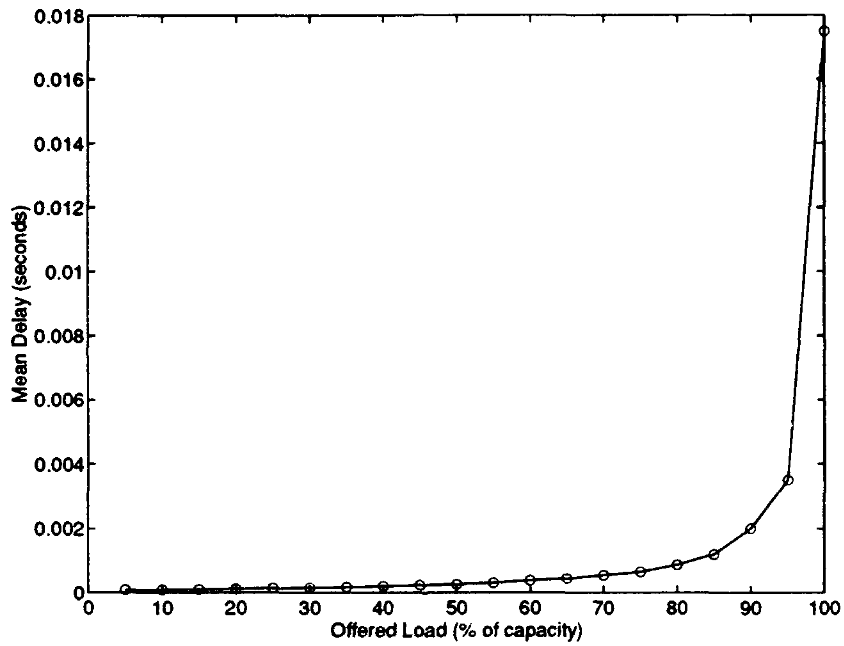

FIGURE 11. Mean Frame Delay vs. Offered Load.

2. CHANNEL UTILIZATION

Figure 12 presents utilization of the channel as a function of the offered load. Utilization increases linearly until the network is saturated, and levels off at approximately

99%. Ulm [Ref. 8] presents a formula for ring utilization as a function of ring latency and average token-rotation time. Figure 12 agrees almost precisely with this theoretical formula.

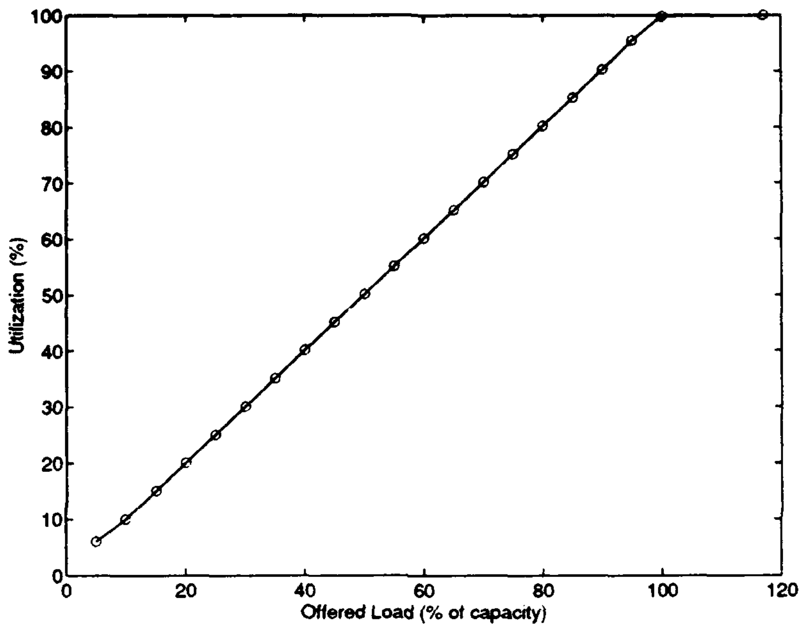

FIGURE 12. ***Utilization vs. Offered Load.***

3. QUEUE LENGTHS

At each node, frames are placed into the transmission queue as soon as they are generated, and they remain there until they are transmitted on the channel. Since our network configuration is homogeneous, for any given offered load, the average number of frames in the transmission queues at the individual nodes are all approximately the same.

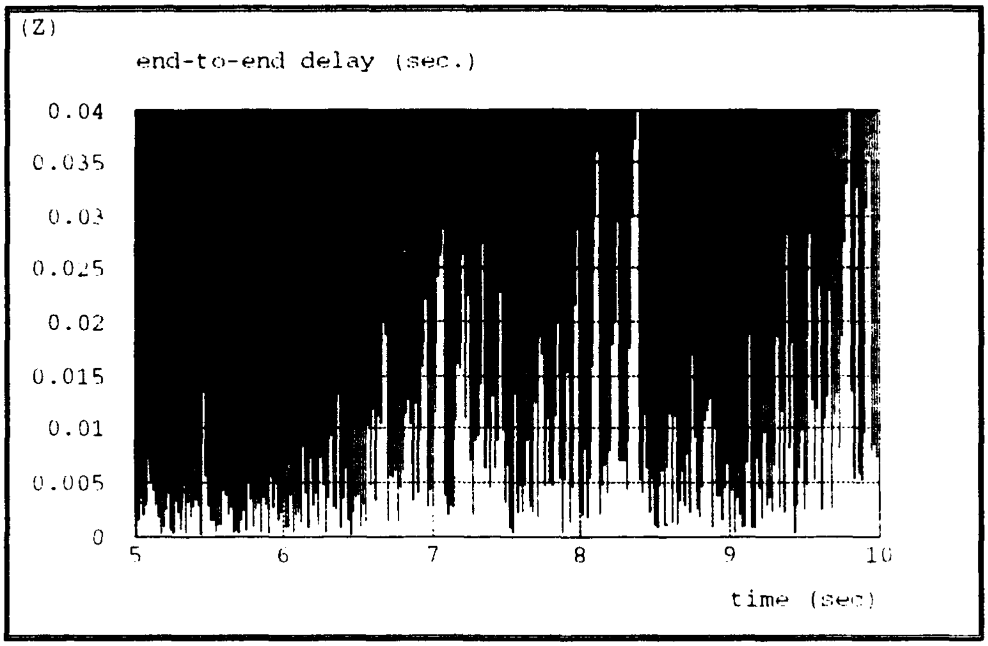

Figure 13. Waiting Time Before Transmission (96% offered load)

The waiting time before transmission for each generated frame for a 96% offered load is shown in Figure 13. We can see that the maximum waiting time is 40 msec. From the simulation results, we have seen that the average queue length is zero, until the offered load exceeds 55%. The average queue length then rises slowly, until at 98% offered load, it is about 44 ms. Even when the offered load is 98%, the maximum queue length is low. Such low queue lengths suggest that even when the offered load is as high as 98% of the capacity, the ring is able to service all the traffic satisfactorily. FDDI timers allow for transmission of multiple asynchronous frames during a single token capture if the preceding token cycle was sufficiently short. As the offered load increases, the frequency with which multiple frames are transmitted also increases.

B. TIMED-TOKEN RING BEHAVIOR

Distinctive features of ring behavior come from restrictions of the timed-token protocol. These features, include the ability to provide support for synchronous applications and the ability to provide equal access to individual nodes for asynchronous transmission.

1. BOUNDED TOKEN-ROTATION TIME

It can be proved that the maximum token-rotation time for any ring configuration is 2 x T_Opr, while average token-rotation time is less than or equal to T_Opr [Ref. 3]. Figure 12 is a graph of average token rotation time as a function of the offered load, using the ring configuration of Table 2, with T_Opr equal 40 milliseconds.

Theoretically, the time required for the token to rotate around the ring in our configuration is 15 microseconds. It is only when the offered load exceeds the capacity of the ring that the average token-rotation time approaches T_Opr as a bound. This indicates that T_Opr value is not a limiting factor for the ring except during burst of activity that temporarily saturate the network. Figure 14 clearly illustrates the asymptotic behavior of average token-rotation time as a function of offered load.

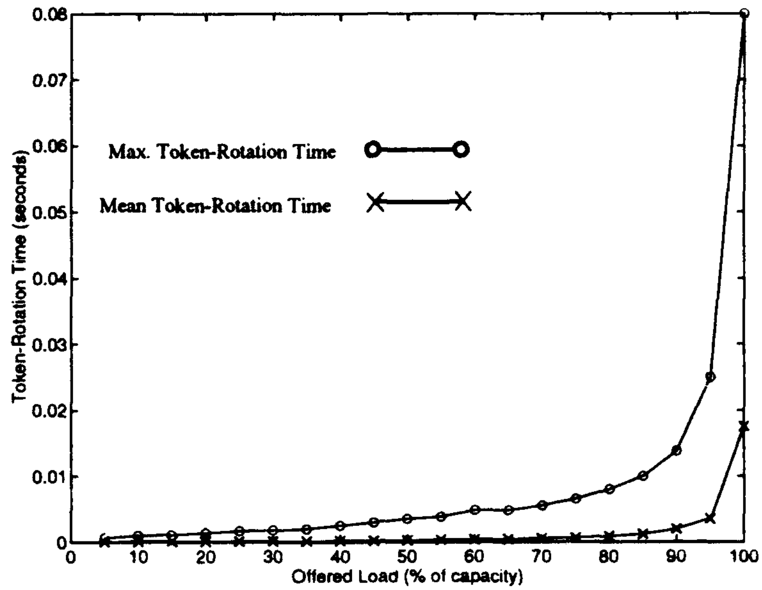

FIGURE 14. Token-Rotation Time vs. Offered Load.

2. SYNCHRONOUS SERVICE

The FDDI MAC protocol guarantees a bounded service interval for synchronous applications. That is, at any given time during ring operation, a node which has been assigned some synchronous bandwidth is guaranteed to be serviced within a specified time interval, which we call the synchronous access time interval. This guarantee is possible only because token-rotation time is bounded above by 2 x T_Opr [Ref. 3]. At ring initialization, nodes negotiate the value of T_Opr, so that synchronous channel-access requirements of all the nodes will be satisfied. The smallest requested value is assigned to T_Opr. Then each node is assigned its synchronous-bandwidth allocation.

The total of synchronous assignments must not exceed 100 percent of T_Opr, since this is the total capacity of the ring. Hence, the value of T_Opr determines the maximum possible volume of synchronous traffic that can be supported in a particular ring configuration. For this reason, it is worthwhile to determine the largest possible value of T_Opr that will maximize the channel access to support the network's traffic. Since the average token-rotation time approaches T_Opr as the offered load nears capacity, clearly T_Opr must be assigned a value less than or equal to the length of the synchronous-access time interval, or synchronous frame delay bound could be violated frequently.

3. FAIRNESS OF ACCESS FOR ASYNCHRONOUS TRAFFIC

Asynchronous traffic is transmitted only if the load on the ring is light enough to support it. According to the standards [Ref. 7], the FDDI MAC protocol "supports fair access at a frame granularity" for asynchronous transmissions.

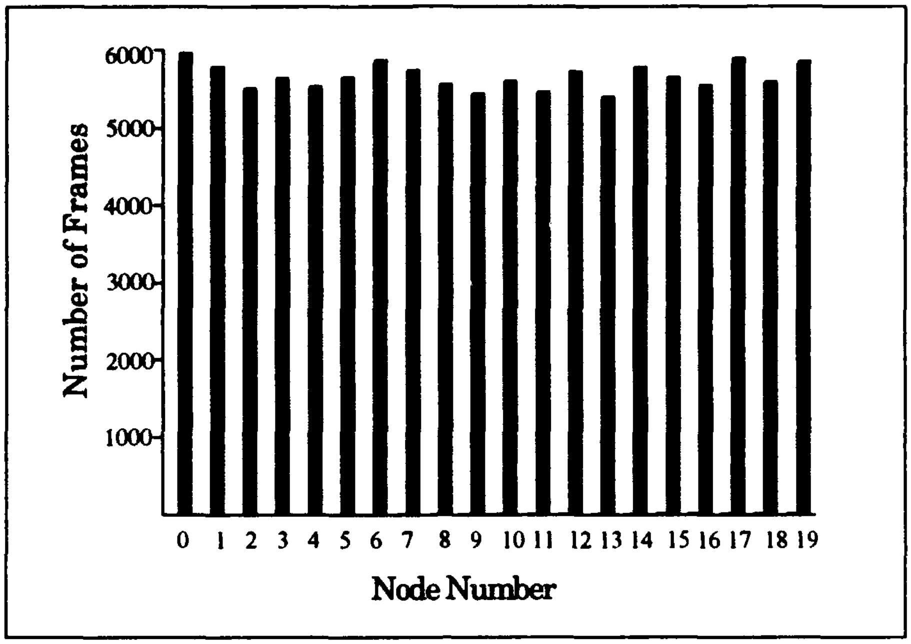

FIGURE 15. Number of Asynchronous Frames Transmitted by Individual Nodes (homogeneous ring).

A theoretical example was presented on Chapter II of this thesis, containing some simplifying assumptions, such as all the nodes have equal access to the channel to transmit asynchronous frames. Simulation data to support the theory of fairness of channel access is most easily obtained by constructing scenarios in which the offered load exceeds the capacity of the ring. Figure 15 is a histogram of the number of frames transmitted by each of the nodes in a twenty-station homogeneous ring where all the traffic is asynchronous and the combined arrival rate of the stations saturates the capacity of the ring (transmission queues always contain frames waiting for transmission). Statistics collected over ten seconds of ring operation show that the largest number of frames transmitted by a single node was 5954 , while the smallest number of frames transmitted by a single node was 5530, 92.87 % of the larger number. Ring operation in this scenario is essentially time-division multiple access (TDMA), with a six-frame time slot for each node during each token rotation. This represents, of course, the most efficient utilization of the channel in a saturated ring.

4. TARGET TOKEN ROTATION TIME

The ring latency consists of the signal propagation delay in the fiber, and the sum of the station latencies. We use a propagation delay of 5.085 μs/km, and a station latency of 0.6μs per station to obtain the results presented in this thesis.

Figure 16 illustrates the effects of the target token rotation time and the number of *actively transmitting stations on the maximum total throughput, for a 20 km ring with* 20 connected stations and a frame length of 1.6 Kbytes. The results where obtained using equation (1) and verified using the simulation model (the differences resulting always less than one percent). Figure 16 also shows the result obtained using Ulm's formula [Ref. 8] which is an upper bound on the maximum throughput since it corresponds to the case of an infinite number of active stations. These results indicate that the OPNET model yields very good approximations for the maximum total throughput.

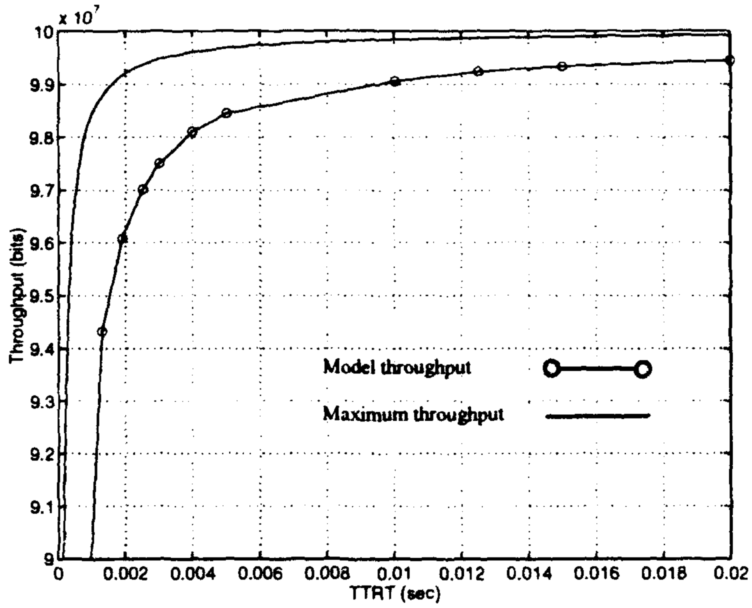

FIGURE 16. Maximum Throughput vs. TTRT for n=20.

V. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

In this thesis we have presented a performance analysis of the FDDI token ring protocol. Using standard measures of performance, we have shown that the average frame delay is low until the ring reaches saturation; utilization and throughput follow the ideal curves, increasing linearly until the ring reaches saturation and then keeping the value. Transmission queue lengths remain small, until the ring reaches saturation, indicating that frames are transmitted almost as soon as they are generated for our particular configuration.

In addition to the standard measures of performance, simulation data is presented to support results from earlier analytic studies of distinctive features of the FDDI protocol. First, we demonstrated the asymptotic behavior of average token-rotation time as a function of offered load. Then we investigated synchronous-frame delays with a more relaxed setting of T_Opr than the specified in the standards. Our results indicate that the service provided when T_Opr is set equal to the desired synchronous-access time interval would be satisfactory for applications which can tolerate excessive delays for a small percentage of synchronous frames. We presented simulation data to demonstrate that the pattern of channel access for nodes transmitting asynchronous traffic is fair, under some simplifying assumptions.

Finally we analyze the influence of the TTRT over the throughput when only one asynchronous priority level is in use. The results presented, reveals a fundamental property of the FDDI protocol: there exists a tradeoff between its efficiency and the effectiveness of its priority mechanism. We have seen that the maximum total throughput remains high as long as the target token rotation time is large with respect to ring latency. However, if even one application on the ring has a very short transmission-delay requirement, the total throughput may be limited, even if the application is rarely active because the TTRT must be set to one half of the maximum acceptable delay. To take advantage of the full effectiveness of the FDDI protocol, efficiency may have to be sacrificed.

Comparisons between the simulated and theoretical results, validate the accuracy of the OPNET simulation tools, demonstrating that it may be used to model other protocols of particular interest.

B. RECOMMENDATIONS

Because of the complexity in developing a general protocol profile to be used to study the communication networks between maritime services which include end-to-end interconnection through the terrestrial network, it is necessary to continue the simulation of other standard protocols and the implementation of new ones.

The development of an efficient gateway capable of satisfying the communication requirements described in chapter one of this thesis is left for further study.

LIST OF REFERENCES

1. Burr, W. E. "The FDDI Optical Data Link," *IEEE Communications Magazine*, Vol. 24, No.5, pp. 18 - 23, 1986.

2. Dykeman, D. and Bux, W., "Analysis and Tuning of the FDDI Media Access Control Protocol," *IEEE Journal on Selected Areas in Communications*, Vol. 6, No.6, pp. 997 - 1010, 1988.

3. Johnson, M.J., "Proof that Timing Requirements of the FDDI Token Ring Protocols are Satisfied," *IEEE Transactions on Communications*, Vol. 35, No.6, pp. 620 - 625, 1987.

4. Johnson, M. J., "Analysis of FDDI Synchronous Traffic Delays," *Proceedings of Systems Design and Networks Conference*, IEEE, 1988.

5. Ross, F.E., "FDDI - a Tutorial," *IEEE Communications Magazine*, Vol. 24, No.5, pp. 10-17, 1986.

6. Stalling, W., "*Data and Computer Communications, third edition*," Macmillan Publishing Company, 1991.

7. Stalling, W.,"*Local and Metropolitan Area Networks, fourth edition*," Macmillan Publishing Company pp. 219 - 272, 1992

8. Ulm, J. M., "A Timed Token Ring Local Area Network and its Performance Characteristics," *Proceedings 7th Conference on Local Computer Networks*, IEEE, pp. 50 - 56, 1982.

9. OPNET, *Example Models Manual,* Mil 3 Inc. 3400 International Drive, N.W. Washington, D.C. 20008, Vol. I, 1991.

INITIAL DISTRIBUTION LIST

1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5101	2
3.	Chairman, Code EC Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5121	1
4.	Professor Shridhar Shukla, Code EC/Sh Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5121	1
5.	Professor Gilbert Lundy, Code CS/Ln Department of Computer Science Naval Postgraduate School Monterey, CA 93943-5000	1
6.	Professor Paul Moose, Code EC/Me Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5121	1
7.	Lt Aldo Schenone Las Orquideas 215, Urb Viru Bellavista, Callao Lima, Peru	1